Fishnet v8 rev 2 Protocol Specification

Introduction

This document describes all the fishnet family of protocols, including frame format (where applicable) and behaviors (e.g., timeouts). All multi-byte fields must be in network byte order.

Fishnet Layer 2 Header

The fishnet L2 header format is depicted in the following table:

Byte Offset							
0	1	2	3	4	5	6	7
Destination L2 Address						Src L2	
Source L2 Address (cont)				Checksum		Length	
L3 Protocol							

Table 1: Diagram of the 16 byte fishnet L2 packet header.

Fishnet L2 addresses are 6 bytes long. The datatype `fn_l2addr_t` is defined in `fish.h` to store L2 addresses. The destination and source address in the L2 header is the immediate destination / source's L2 address. The L2 address F-FF:FF:FF:FF:FF:FF is the broadcast address. The L2 address F-00:00:00:00:00:00 is reserved and is invalid.

The `Checksum` field is computed using the Internet checksum algorithm. It covers the entire L2 frame, from the first byte of the destination address to the last byte of the payload, inclusive.

The `Length` field contains the length, in bytes, of the entire L2 frame. The length starts at the first byte of the destination address to ends at last byte of the payload, inclusive.

The `L3 Protocol` field contains the protocol of the L3 frame. There are currently four define L2 protocols as listed below:

1	Fishnet L3 Protocol
2	Fishnet ARP Protocol
3	Fishnet Neighbor Protocol
4	Fishnet Distance Vector Routing Protocol

Table 2: Fishnet L2 Protocol Numbers

Protocol details are provided later in this document.

Fishnet Layer 3 Header

The fishnet L3 header format is depicted in the following table:

The `TTL` indicates the time-to-live value for the packet. The largest legitimate TTL value is defined by the constant `MAX_TTL`. The TTL is decremented by one every time a frame traverses a network hop. When

Byte Offset							
0	1	2	3	4	5	6	7
TTL	Protocol	Packet ID				Src L3	
Src L3 (cont)		Destination L3 Address					

Table 3: Diagram of the 14 byte fishnet layer 3 packet header.

the TTL reaches zero and the frame hasn't reached its destination, the frame is dropped and a FCMP error response is generated. See the section describing FCMP for more details on this error message.

The `Protocol` field indicates which protocol L4 is carried in the packet. The known values for this field, and their associated interpretations, are:

1	Packet Protocol (undocumented, but don't drop)
2	Echo Protocol
3	Naming Protocol
4	Link State Routing Protocol (undocumented, but don't drop)
5	Transport Protocol (undocumented, but don't drop)
6	FCMP Protocol

Table 4: Fishnet L3 Protocol Numbers

Protocol details are provided later in this document.

The `Packet ID` field contains a unique number that the original sender assigns to the packet. The only constraint on the value is that there can only be one packet on the network with a particular sender-packet ID pair. The libfish function `fish_next_pktid` is used to generate these numbers.

The source and destination L3 address are of type `fnaddr_t`, which is defined in `fish.h`. This is a four byte address, and is printed using a notation very similar to IP address's dotted-quad notation. The source and destination addresses are L3 address – they indicate the original packet source and the final packet destination. These address do not change as a packet is forwarded through the network.

Fishnet support L3 broadcasts that flood the network. That is, a packet addresses to the L3 broadcast address (F-255.255.255.255) is seen at every node on the entire fishnet.

Fishnet Layer 3 Protocols

This section describes each of the fishnet layer 4 protocols.

2 – ARP

Fishnet employs an ARP L4 protocol to resolve the L3 addresses into L2 addresses. The ARP header is:

The `Query Type` field indicates the type of the ARP packet. The two legitimate values are:

- 1 – ARP Request
- 2 – ARP Response

Byte Offset

0	1	2	3	4	5	6	7
Query Type				Queried L3 Address			
L2 Address for the Queried L3 Address							

Table 5: Diagram of the 14 byte fishnet ARP packet header.

The `Queried L3 Address` is the L3 address being resolved. It must be filled in for both requests and responses. The third field, `L2 Address for the Queried L3 Address`, is the L2 address the L3 address resolves to. This field is only valid in response packets.

The L2 destination for ARP requests is the corresponding broadcast L2 address. The responses are unicast back to the requestor.

ARP mappings are cached for 180 seconds, after which another ARP request/response sequence is necessary to resolve the L3 address again. A resolution is attempted exactly 4 times, with 2.5 seconds in between attempts. After the fourth non-response (which will occur 10 seconds after the first request), the L3 address is assumed to be unreachable and all queued frames for that address are dropped.

3 – Neighbor Protocol

Fishnet has no inherent ability to provide a list of neighbors to each fishnode, nor can it tell a node when a particular neighbor may join or leave the network. The neighbor protocol must be employed by each fishnode to track this information. The protocol allows nodes to discover new neighbors and identify when a neighbor leaves the network.

The protocol is a simple, two step protocol. First, a fishnode broadcasts a neighbor discovery packet. This packet is only received by the neighbors of the fishnode. The discovery packet must be sent to the L2 and L3 broadcast addresses using a TTL of 1. Setting the TTL to 1 prevents the frame from flooding the network.

Upon receipt of a discovery request, a fishnode *unicasts* a discovery response back to the fishnode that initiated the discovery. The unicast frame must also use a TTL of 1. Using this technique a fishnode can discover its neighbors. The neighbor protocol packet is as follows:

Byte Offset

0	1	2	3	4	5	6	7
L3 Address				Packet Type			

Table 6: Diagram of the 6 byte fishnet neighbor packet header.

The `L3 Address` field contains the L3 address of the node that sent the neighbor packet.

There are two possible values for the `Packet Type` field:

- 1 – Discovery Request
- 2 – Discovery Response

The neighbor protocol has no provision for directly "discovering" when a neighbor leaves the network. This detection is accomplished through timeouts. If a neighbor has not been heard from within a set amount of time, that neighbor is assumed to have left the network. This detection method requires ongoing, periodic probing of the network. A fishnode must probe the network with neighbor requests once every 30 seconds. If a neighbor is not heard from within 2 minutes the neighbor is assumed to no longer exist.

4 – Distance-Vector Routing Protocol

The fishnet distance vector protocol is one of the routing protocols available in the for maintaining the routing tables. Fishnet's distance-vector routing is a vanilla DV routing protocol, with the following optimizations:

- Split-Horizion with Poison-Reverse. When node A shares its table with node B, all entries that use node B as the next hop have a metric of infinity.
- Routes can be "withdrawn" by the node that originally advertised it by specifying a metric of infinity.
- Nodes keep a complete history of fallback routes to use when a route is withdrawn or a neighbor leaves the network.
- Route updates (announcing new routes or withdrawing stale routes) are triggered (before the normal 30 second periodic update) when there is a material change to the route advertisement (different metric or a new advertisement). **Note:** when a triggered update sends a routing table the 30 second time for the periodic table broadcast should be reset.

The last three optimizations above increase route convergence time with a change is detected. The first optimization allows a fishnode to tell its neighbors that a previously working route is now invalid. This withdraw spreads through the network much faster than waiting for each individual fishnode to time out a route. The third optimization increases the speed with which backup routes are discovered. Keeping a complete history of active routes minimizes the routing disruption due to topology changes.

The format of a distance vector routing packet is:

Byte Offset							
0	1	2	3	4	5	6	7
L3 Address				# of Advertisements		0 or more route advertisements	

Table 7: Diagram of the 6 byte fishnet distance-vector routing protocol header.

The `L3 Address` field contains the L3 address of the node that sent the neighbor packet.

The `# of Advertisements` in the routing header indicates the number of advertisements that are contained in the packet. The format for an advertisement is:

Byte Offset							
0	1	2	3	4	5	6	7
Destination Address				Netmask			
Metric							

Table 8: Diagram of the 12 byte fishnet distance-vector route advertisement.

The distance vector protocol uses hop-count as its routing metric. A value of `MAX_TTL` is interpreted as infinity, and indicates that particular destination is unreachable. Due to the MTU restriction, it is possible the entire routing table will not fit into a single distance vector frame, so every route need not be advertised in every routing packet. A route must be advertised once every 30 seconds. If the next-hop router has not advertised a route in the previous 3 minutes, the route should be marked as stale and withdrawn by advertising it with a metric of infinity for 3 additional minutes.

The distance vector protocol discovers a nodes neighbors (nodes that send it a routing update) in addition to multi-hop routes. Because of this it is unnecessary to use both the neighbor protocol and distance vector routing. In addition, since split-horizion with poison-reverse unicasts the route advertisements to each neighbor, a broadcast advertisement containing 0 routes must be sent once every 30 seconds to facilitate neighbor detection.

All Fishnet nodes implementing the DV protocol much be compatible with each other. This is normally achieved by ensuring a node can receive DV packets that are both broadcast and unicast to it. When implementing the protocol, don't make any assumptions that break compatibility unless the specification explicitly states otherwise.

Some important items to note while implementing DV:

- Only the best route to a destination should be in the FIB. If the best route changes remove the old one and insert the new one into the FIB. If there is no known route there should be nothing in the FIB for that destination.
- A route gets invalidated after 180 seconds without an advertisement.
- To implement route history you need to track all routes in your RIB, even the ones that have a worse metric. This includes tracking the invalid / timeout state for all routes.
- When the active / best route times out, fall back to the 2nd best route in your route history (if implemented). If there isn't one remove the route from the FIB.
- If you are implementing route withdraws send out 180 seconds of "infinite" metric only when there is no route to the destination. If you are implementing route history this includes looking for a 2nd best route prior to initiating the withdraw.
- If you are implementing route withdraws only withdraw routes that you has previously advertised.

Fishnet Layer 4 Protocols

2 – Echo Protocol

The echo protocol has three types of packets, two flavors of requests and a response. When a node receives the echo request it copies the payload into an echo response packet and sends it back to the original source. The echo protocol header is:

Byte Offset							
0	1	2	3	4	5	6	7
Echo Code				Payload...			

Table 9: Diagram of the 4+ byte fishnet echo packet.

There are three possible values for the `Echo Code` field:

- 1 – Echo Request
- 2 – Echo Response
- 3 – Echo Request, with a Spoofed Response Address

The first two codes, 1 and 2, are straightforward. When a node receives an echo request addressed to it, the payload is copied into an echo response packet and sent back to the requester.

The third code, request with spoofing, is similar. When a node receives a type 3 echo packet addressed to it, the payload is copied into a type 2 response and sent back to the requester, but one twist. The requester's L3 address is used as both the source *and* destination L3 address in the response frame. This feature is used to enable traceroute to separately identify both the forward and reverse paths taken by a packet through the fishnet.

3 – Naming

A fishnode can optionally give itself one (or more) human-readable names. A name is a string of printable, non-whitespace, ASCII characters no more than 128 characters long. Names are resolved into L3 addresses using the naming protocol.

The naming protocol operates by flooding the L3 network with a lookup request, ensuring all nodes in the network receive the request. The node that has the name then sends a unicast response back to the requesting node. There is no provision for detecting or dealing with duplicate names. The format of a name resolution request/response is:

Byte Offset

0	1	2	3	4	5	6	7
Query Type				Response L3 Address			
TTL (s)		Queried Name ([1-128] bytes, no null termination)					

Table 10: Diagram of the fishnet naming packet.

The `Query Type` field indicates the type of the naming packet. The two legitimate values are:

- 1 – Name Resolution Request
- 2 – Name Resolution Response

The `Queried Name` is the name being resolved. It must be filled in for both requests and responses. The remaining fields, `Response L3 Address` and `TTL` are only valid in response packets. `Response L3 Address` is the L3 address the name resolves to. `TTL` is the amount of time, in seconds, which the name response should be cached.

The request packet is flooded on the network by sending it to the L3 broadcast address with the TTL set to MAX_TTL.

A resolution is attempted exactly 4 times, with 5 seconds in between attempts. After the fourth non-response (which will occur 20 seconds after the first request), the name is assumed to be non-existant.

6 – FCMP

FCMP messages are generated by the fishnodes in response to certain network events. They carry diagnostic information another fishnode/application can use to help determine what whet wrong with a particular transmission. Fishnodes are required to generate these packets as appropriate. The format of a FCMP packet is:

Byte Offset							
0	1	2	3	4	5	6	7
Error Code				Sequence Number			

Table 11: Diagram of the 8 byte FCMP packet header.

FCMP packets are always generated in response to an error condition with a *unicast* frame (the original packet). The `Sequence Number` field in the FCMP header contains the value of the `Packet ID` field in the original packet. The L3 destination of the FCMP packet must be the L3 source of the original packet. FCMP packets are **never** generated in response to a broadcast frame, in response to another FCMP frame, or in response to a frame with a broadcast source address.

The valid FCMP error codes are:

- 1 – TTL Exceeded:
  This FCMP type is generated with the TTL field is decremented to 0 *and* the packet hasn't reached its final destination.
- 2 – Network Unreachable:
  This error is generated when a frame is dropped because there is no route to the destination. This is an L3 forwarding failure.
- 3 – Host Unreachable:
  This error is generated when a frame is dropped due to an ARP lookup timeout. This is an L2 error.